(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 May 2001 (17.05.2001)

PCT

(10) International Publication Number
WO 01/35256 A2

(51) International Patent Classification[7]: G06F 17/00

(21) International Application Number: PCT/US00/30784

(22) International Filing Date:
9 November 2000 (09.11.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/164,170 9 November 1999 (09.11.1999) US

(71) Applicants and
(72) Inventors: **TUCKER, Brent, David** [US/US]; 9319 South Sand Hill Way, Highlands Ranch, CO 80126 (US). **KOOPMANN, James, Frederick** [US/US]; 2233 Ponderosa Road, Franktown, CO 80116 (US).

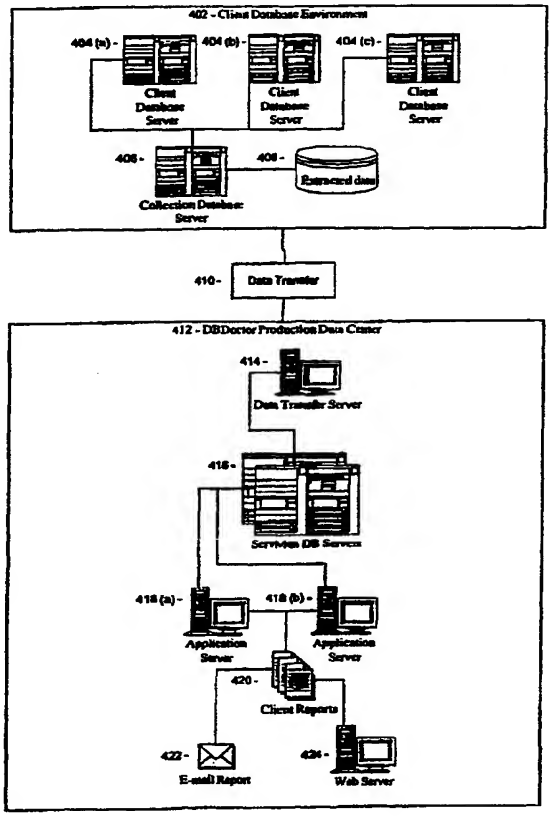
(74) Agents: **BARAN, Alexandra, J.** et al.; Cooley Godward LLP, Patent Group, 11951 Freedom Drive, One Freedom Square - Reston Town Center, Reston, VA 20190-5601 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: SYSTEMS AND METHODS FOR COLLECTING, STORING, AND ANALYZING DATABASE STATISTICS



(57) Abstract: A system and method for collecting, storing, retrieving and analyzing database statistical information. A collection agent on a database server interrogates the database on a defined interval to determine the database state, and to derive statistics about database performance, structures, and usage. This statistical information is stored within a collection database as historical data. Periodically, the historical data is unloaded from the collection database, transferred via a Transfer Protocol to a data warehouse server, and loaded into said server. Summarization of the data, and Trends-based analysis, are performed against the data with results stored in the data warehouse as persistent storage. Periodically, an application server interrogates the data warehouse for the database being analyzed, and produces a series of reports which show database trends, reliability, and performance information in graphical, textual, and tabular formats.

WO 01/35256 A2

Published:
— *Without international search report and to be republished upon receipt of that report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEMS AND METHODS FOR COLLECTING, STORING, AND ANALYZING DATABASE STATISTICS

FIELD OF THE INVENTION

[0001] This invention relates generally to the field of database administration and, more specifically, to systems, methods, and applications that provide trending analysis of a database system by monitoring internal statistics and structures as provided by a database vendor or through custom process and data acquisition techniques, as such information spans time. This invention also relates to systems, methods, and applications that use collected information concerning databases to assist with database analysis, including, for example, performance evaluation, reliability analysis, and management of database systems.

BACKGROUND OF THE INVENTION

[0002] A database system is a product that resides on one or more computer systems and provides multiple users with access to data. As databases are used, methodologies for accessing stored data, optimizing database processes, and storing parameter requirements change. It follows that complex database systems may require extensive management. Managing database systems includes the monitoring of processes and resources in order to detect deterioration in the overall reliability and efficiency of a database system.

[0003] Parameters that can be monitored within a database include, but are not limited to, CPU utilization, physical storage utilization, memory utilization, network traffic, physical database structures, logical database structures, referential integrity, and parameters and files which are required for reliability. By interrogating the values of these parameters on a timed schedule, storing historical information, and using that information to determine trending patterns, monitoring of that information can help to predict anomalies within the system, enhance reliability, and increase the performance of a database. Trends may be evaluated over any period of time, including hours, days, weeks, or months. By retaining historical data for a relatively long period of time, the trends shown will be more accurate, and may be more useful to a database

administrator. By reviewing trend information, insight may be gained into both normal and abnormal behavior of a database system.

[0004] The many benefits that flow from active monitoring of a database system include, for example, detection of potential or actual problem areas within a database system; location of intermittent problem areas; identification of increased periods of load on the database system; verifying the overall quality and performance of a production system; and alleviation of the time and resource constraints placed on database administration personnel. When errors or degradation of performance trends occur, repairs may begin prior to complete failure of the database environment.

[0005] Trends-based analysis is powerful due to the predictive nature of such analysis. By reviewing growth patterns, utilization patterns, and performance patterns, one may derive the point at which a database may begin to fail to meet the needs of a process. Once a trend is identified, resources and processes may be modified to alleviate problems associated with the trend.

[0006] Database servers may be run on many different hardware and operating system platforms. While each of these platforms has unique features, database servers are common across all platforms. By monitoring multiple database servers on non-heterogeneous platforms, and providing a common reporting mechanism, a framework of communication can be established that allows personnel to perform more efficiently.

[0007] Goals of trends-based analysis of database systems are to reduce expenses, improve performance and availability, to sustain or increase revenue, and to improve the performance of personnel by alleviating their need to continuously monitor the database through manual means.

[0008] Many difficulties are present in existing approaches. First, manual monitoring processes are error prone. Since a human interface is required to perform multiple steps for data collection, and then analyze resulting data and interpret the results, the analysis and decision making process may not be consistent. An inconsistent method of obtaining results may result in improper decisions being made concerning database administration. This problem is compounded by more than one person attempting the analysis, since past experience invariably skews the interpretive results.

[0009] Manual monitoring is also labor intensive. For each step of the process, scripts must be developed to interrogate a specific element of the database. These

scripts must be run in a prescribed order, and the results must be tabulated and compared in order to understand the current state of the database. Additionally, analyzing the trend data is extremely difficult, and the results of the analysis may not be accurate.

[0010] As database systems are used, the requirements of the underlying hardware and software are compounded by the fact that the number of users, the amount of data, and the number of processes run against a database are continually increasing. Additionally, the use of database systems has grown beyond traditional boundaries, and databases are being used in more areas of industry and with more types of applications. As such, requirements for monitoring and correcting database problems have become more diverse.

[0011] With database systems found in numerous aspects of government and industry, it would be desirable to provide a common set of tools and processes that provide an automated and complete analysis of database systems. Such a set of tools and processes would be valuable to database administrators and the clients they represent.

SUMMARY OF THE INVENTION

[0012] In one innovative aspect, this invention is directed to a database system statistic collection, analysis, repository creation, and consistent reporting tool that monitors database systems and produces textual, tabular, and graphical reports describing both Trends-based and static data. In one exemplary embodiment, the tool utilizes a database system for collection, standard Internet protocols for file transmission, a repository database for warehousing of data, and reporting tools which may be viewed electronically, via a web browser or other display means, or as a paper based report. Additionally, clients receiving the report may use a web browser or other query device or software to drill down through data that is retained on the data warehouse server.

[0013] The database servers to be monitored may reside in local or remote data centers, and each data center may house one or more database systems.

[0014] Preferably, the data center housing the analysis system consists of one or more database servers, one or more application servers, and other peripheral equipment as

needed. Multiple data centers may exist. Data can be retained on the database server(s) for a period of time, allowing trends analysis to be more accurate. All incoming files can be written to a permanent storage device as an archival method.

[0015] The analysis process may consist of one or more programs specifically designed to perform trends-based analysis, object analysis, structure analysis (of database structures), and reliability analysis. Additionally, such processes may summarize data and provide categorization for further reporting.

[0016] The reporting process queries the analyzed and summarized data warehouse for trend information, potential points of contention within the database, performance problems, and reliability concerns. The resulting data is formatted into a consistent series of reports including, but not limited to, executive summaries, management reports, progress reports, reliability reports, and trends-based proactive recommendations and descriptions of potential problems.

[0017] The statistics to be collected reside on one or more database systems, with each system storing its own statistics collection individually, or optionally with all collection of information written back to a central collection point. Statistics are collected on some predefined interval, are converted through selected algorithms to a historical record format, and are stored in a database schema designed specifically for the system.

[0018] Each evaluated database system has a database engine, an underlying operating system, and a designated storage facility for the collection methods. Data collection scripts determine which data to collect, the validity of the data, and ensure that the data is stored correctly. A scheduler provided by the operating system controls the initialization or termination of the collection scripts at the predetermined interval. The host for the database has a clock, which is used for time stamping the historical data to preserve order of entry into the system.

[0019] Data transport occurs on each computer that hosts a collection process, and sends the collected data back to the analysis environment via the Internet. The protocol used for this transmission will vary within commonly used data transmission protocols. As data is received in the analysis environment, it is written to permanent off line storage, for example a CD-ROM, so that the data will be available in the case of

disaster recovery, or if a client should need the data again after a retention period has expired.

[0020] The data is received in the analysis environment and is loaded into a staging area. A conversion process uses the staged data to load the data into the data warehouse. A summarization and analysis process interrogates the data, providing several levels of summarization and performing Trends-based analysis on the data.

[0021] Data retention is specific to the needs of the owner of the database system being analyzed. All collected data, in both raw and summarized form, are retained within the data warehouse. A remote user of the system may connect to the data warehouse server via the Internet and run other analysis against the data as needed, or retrieve additional reports that are not included in the standard report package.

BRIEF DESCRIPTION OF THE DRAWINGS

[0022] Figure 1 is a flow chart illustrating a process in accordance with the present invention for collecting statistics, analyzing those statistics, and effecting reporting within database systems.

[0023] Figure 2 is a flow chart illustrating a collection process flow in accordance with the present invention.

[0024] Figure 3 is a flow chart illustrating a reporting process flow in accordance with the present invention.

[0025] Figure 4 is an illustration of a database system architecture that may have an application in accordance with the present invention configured thereon.

[0026] Figures 5(a)-(f) are screen snapshots showing sample reports in accordance with the present invention.

DETAILED DESCRIPTION

[0027] Turning now to the drawings, Figure 1 is a flow chart illustrating a process in accordance with the present invention for collecting statistics, analyzing those statistics, and effecting reporting within database systems. More specifically, Figure 1 describes an overall process 100 that provides for (1) the collection of data, (2) local storage of that data on a database to be evaluated or a repository database established for evaluation, (3) the transport of that data to a remote server or servers, and (4) the

loading, summarization, analysis, trend based analysis, storage, extraction and reporting of that data. At step 102, the process flow can be entered through software at an evaluation site. At step 104, the collection mechanism is started as a timed activity on the database server holding a repository. One or more collection mechanisms may be started depending on the number of databases to collect from. At step 106, upon completion of the collection of data, and at a pre-determined interval, the data is extracted from the database and placed into a standard export format. At step 108, the system performs an integrity check to determine if the extraction was successful. At step 110, if the extraction process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 112, if the extraction process does succeed, the exported data is transferred to a production data center. At step 114, the system performs an integrity check to determine if the transfer of data was successful. At step 116, if the transfer process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 118, if the transfer process does succeed, the data from the client database is loaded into a staging area in the DBDoctor database server. At step 120, the system performs an integrity check to determine if the data load was successful. At step 122, if the data load process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 124, if the data load process does succeed, the data from the client database is summarized through the execution of a series of summarization scripts. At step 126, the system performs an integrity check to determine if the summarization was successful. At step, 128, if the summarization process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 130, if the summarization process does succeed, the data from the client database is analyzed for trends and other criteria through the execution of a series of analysis scripts. At step 132, the system performs an integrity check to determine if the analysis was successful. At step 134, if the analysis process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 136, if the analysis process does succeed, reports about the data are generated in multiple formats, including reports at executive, managerial and technical personnel levels, for delivery via email, PDF format, or other delivery

mechanism. At step 138, the system performs an integrity check to determine if the reporting was successful. At step 140, If the reporting process was not successful, the process flags the error in the repository database and branches execution to item 144, End of Process. At step 142, If the reporting process does succeed, the data warehouse is updated to indicate the completion of execution for the currently processed database system. At step 144, the end of the process occurs.

[0028] Figure 2 is a flow chart illustrating a collection process flow in accordance with the present invention. More specifically, Figure 2 describes a process 200 for a Statistic and Structure Collection Process in accordance with the present invention. In general, this process determines the number and types of database environments from which data will be collected, collects the data by processing each database, ensures the validity of the collected data, and transfers the data to the remote evaluation site.

[0029] At step 202, the process flow can be entered through software at the evaluation site. At step 204, the collection mechanism is started as a timed activity on the database server holding a repository. One or more collection mechanisms may be started depending on the number of databases from which to collect. The databases to be evaluated are determined and placed into a queue. At step 206, the next database to be processed on the queue is selected for collection. At step 208, the system performs an integrity check to determine if the end of queue condition was reached. At step 210, if the end of queue condition was encountered, the process checks to see if the data has been flagged for transfer to the remote analysis data center. At step 212, if the data has been flagged for transfer, the data is extracted from the database repository at the client site, and is transported to the DBDoctor data center specified by the operations staff of the licensing company. At step 214, the process ends. At step 216, if the end of queue condition was not encountered, the system performs an integrity check to determine if the database to be collected is accessible. If the database to be collected is not accessible, process control is returned to item 206, 'Get next database on the queue to collect'. At step 218, if the database to be collected is accessible, variables are set which control the collection process for the database to be collected. At step 220, the database type and version are determined by the process, and variables describing these values are set for further processing. At step 222, a queue is created which contains the internal structures to be collected. At step 224, the next structure to be collected is

obtained from the queue defined in item 220. At step 226, the system performs an integrity check to determine if the end of the collection queue has been reached. If the end of the collection queue has been reached, the process branches execution to item 206, 'Get next database on the queue to collect'. At step 228, if the end of the collection queue has not been reached, structure statistics are collected for the current evaluation database type and version. At step 230, the system performs an integrity check to determine if a valid collection has been taken. At step 232, if collection does not succeed, a collection error flag is stored in the registry, and control branches to item 224, 'Get next structure to collect'. At step 234, statistics are updated on the repository database. Execution of the process branches to item 206, 'Get next database on the queue to collect' for further processing.

[0030] Figure 3 is a flow chart illustrating a reporting process flow in accordance with the present invention. More specifically, Figure 3 describes a reporting process 300 in accordance with the present invention. In general, this process determines the number of databases to report on, loads a queue of those databases, processes the queue, and generates and delivers reports.

[0031] At step 302, the process flow can be entered through software at the production data center. The reporting mechanism is started as a timed activity on an application server. At step 304, the databases to be evaluated are determined and placed into a queue. At step 306, the next database to be processed on the queue is selected for reporting. At step 308, the system performs an integrity check to determine if the end of queue condition was reached. At step 310, if the end of queue condition was encountered, the process updates the data warehouse with a completion status. At step 312, the process terminates. At step 314, the system performs an integrity check to determine if the data for the current database is available. If the data is unavailable for processing, execution branches to item 316, 'Perform Load'. At step 316, the load process is invoked for the current database. Execution branches to item 322, 'Perform Summary'. At step 318, get information about the client, client databases, and recipient list for processing from the data warehouse. At step 320, the system performs an integrity check to determine if the summarization for the current database has been run. If the data is not summarized, execution branches to item 322, 'Perform Summary'. At step 324, the trend analysis processes are invoked for the current database. At step 326,

management reports are produced for the current database. At step 328, executive reports are produced for the current database. At step 330, detailed data reports are produced for the current database. At step 332, reports are assembled for the current database and prepared for delivery. At step 334, data is made available for Internet based queries. At step 336, the HTML version of the report is generated. The report is stored on the web server. At step 338, the Adobe PDF version of the report is generated. The PDF report is stored on the mail application server. At step 340, the data warehouse is interrogated to determine if the customer has a recipient list, which includes HTML reports. If not, execution branches to item 344, 'Does customer want a PDF report?'. At step 342, an email is dispatched to the customer as an informational message informing them that their report is ready. At step 344, the data warehouse is interrogated to determine if the customer has a recipient list, which includes PDF reports. If not, execution branches to item 348, 'Exit'. At step 346, an email is dispatched to the customer with a PDF file attachment for the current database. Process execution branches to item 306, 'Get next database on the queue to report.

[0032] Figure 4 is an illustration of a database system that may have an application in accordance with the present invention configured thereon. More specifically, Figure 4 describes the architecture and design of the database monitoring system environment. In general, the architecture has two physically separate areas, item 402, 'Client Database Environment', and item 412, 'DBDoctor Production Data Center'. The Client Database Environment will be required for each customer served by this service. The DBDoctor Production Data Center may be replicated as required for scalability. The Client Database Environment 402 consists of one or more of item 406, 'Collection Database Server', and zero or more of item 404, Client Database Server. Item 408 is a temporary file generated during extract processes. The databases 404 to be evaluated are loaded with software that enables them to be evaluated from a remote location. The client may have one or more databases to be evaluated. The database being evaluated may also hold the repository database, depicted by item 406, 'Collection Database Server'. The architecture is flexible and will be designed and installed to the customer specifications. Each database server 406 at the client facility may have one or more databases loaded in order to be evaluated. There is no limit to the umber of databases a customer may have, as demonstrated by items 404 (a), 404(b) and 404(c). The

Collection Database Server 406 is loaded with an application in accordance with the present invention and databases components, together referred to as a repository. This repository preferably resides on a server with a database installed and functional. The repository may reside on an existing production database, or may reside on a server dedicated to this task. Additionally, multiple Collection Database Servers 406 may be installed at each customer site.

[0033] On a pre-defined interval, data is extracted from the repository database residing on item 406, and stored temporarily in a disk file at the customer site pending transfer of the data via item 410, 'Data Transfer'. The data transfer component 410 may make use of any of a number of electronic transfer protocols, including File Transfer Protocol, Hyper Text Transfer Protocol, Secure Copy, or Virtual Private Network transmission. This Data Transfer uses push technology to deliver the data, contained in the Extracted Data file illustrated by item 408, to the Data Transfer Server, represented by item 414 at the DBDoctor Production Data Center.

[0034] The Production Data Center 412 houses the equipment and software required for the analysis, data storage, report generation, and client delivery components required by the service. The data transfer server 414 is a highly reliable component that facilitates the transfer of data from a client site to the Production Data Center 412. This server may be a single component, or may be clustered or placed in another highly available environment as needed. This server supports the receipt of information from multiple transfer protocols.

[0035] The System Database Servers 416 house the data warehouse component of the service. This data is retained for the number of days deemed necessary for the analysis of data to produce trend-based reports.

[0036] The Application Servers 418 are multi-function, serving to facilitate the load process, summarization process, analysis and reporting process, and mail delivery processes. As illustrated, one or more application servers are required for operation of the service. Items 418 (a) and 418 (b) are representative of these servers.

[0037] The Client Reports 420 are one of the end results of the service. They are generated on item 418, 'Application Server', and are then made available to the client for use. The E-Mail report 422 is one of the available versions of the reports generated.

[0038] The Web server 424 makes available the customer data for further analysis. Functions of the web server include, but are not limited to, drill down reporting of data, enhanced trend analysis, shema and structure review, and archival report storage and delivery. This item, as depicted by item 424, may be a single server or one of many as required for load balancing and reliability.

[0039] The collection mechanism uses a tabular replication of structures to be collected. This replication of internal structures preferably is fully aware of differences between versions of databases being collected and will modify itself depending on the differences between those versions. In addition to holding purely structural object data, the collection tables also have various columns to handle items such as customer information, timestamp data, and runtime logic.

[0040] Preferably, the collection mechanism is a collection of tables that are complete replicas of internal database structures. These tables are identical to the collected structures in columns names and data types. In addition to the one to one relationship of column values, customer and timestamp columns may be added. During the collection process a simple select and insert statement may be generated to pull the internal data from the database structures and place that data into the replicated structures. At the time this information is selected, customer data and a timestamp is also placed into the table for each row. The customer data is a mechanism to maintain identity of the data after transport to the warehouse storage facility, and the timestamp is placed into the rows so that an image of how the data changes over time may be obtained, and so that data can be grouped by such means as hours of the day, day of the week, and week of the year. Since there are many different database versions that the collection needs to pull data from, a fully aware environment is utilized that will extract the full data definition of the structures to be collected from the internal database dictionaries of the databases to be collected. Once this definition has been extracted the collection process can dynamically generate the select and insert statements that will be used to populate the replicated structures. In addition to this, if during the collection it is determined that an internal structure has changed, the system can also dynamically generate simple data definition language statements that can be submitted against the database to alter the replicated structures to match the internal structures to be captured.

[0041] Figures 5(a)-(f) are illustrations of reports that may be generated using a database management system in accordance with the present invention. Figure 5(a) is an example of an executive summary report, which includes a list of event categories 510 and event severeties 512. Figure 5(b) provides an illustration of an IT management report that lists events in order of severity, e.g., critical 520, serious 522, warning 524, etc. Figure 5(c) provides an illustration of a detail report that addresses in more detail the events listed in the IT management report, shown in Figure 5(b). Figure 5(d) provides an illustration of an executive summary including a usage chart. Figure 5(e) provides an illustration of a management report including a graphic representation of I/O mount point percentages, and Figure 5(f) provides an illustration of detail report regarding mount points.

[0042] Because the invention is susceptible to various modifications and alternative forms, specific examples thereof have been shown in the drawings and are herein described in detail. It should be understood, however, that the invention is not to be limited to the particular forms or methods disclosed, but to the contrary, the invention should encompass all modifications, alternatives, and equivalents falling within the spirit and scope of the appended claims.

WHAT IS CLAIMED IS:

1. A system for collecting, storing, analyzing, and reporting statistical and historical data within a plurality of database systems, comprising:

a series of data collection scripts resident on a plurality of database servers, each said script including statistic collection commands and process invocations that are native to at least one of an operating system on which they are run and a database system on which they are run;

a file transport process, which delivers data collected by the data collection scripts to an analysis site;

a load process, which places the collected data into at least one database repository;

a series of analysis and summarization scripts, resident on the database repository server, data warehouse server, or an application server, which may be used to provide historical trends-based analysis of the data collected;

at least one application server, coupled to the database repository, which provides an interface to the reporting system, and provides conversion of the report into multiple formats, including, but not limited to, PDF and HTML;

at least one email server, electronically coupled to the application server(s), for transmitting the reports to the client requesting the information; and

at least one web server, electronically coupled to the application server(s) and database repository server(s), for providing the client with additional information or access to HTML based reports.

2. A method of database analysis, comprising the steps of:

collecting data and statistics on one or more databases that reside on a plurality of database servers;

storing said data and statistics;

transporting said data and statistics to an analysis site;

placing said collected data and statistics into at least one database repository;

providing historical trends-based analysis on said collected data and statistics;

converting results of said analysis into one or more alternative formats;

transmitting said results of said analysis to a recipient; and

providing, to said recipient, additional information about said collected data.

3. A method as in claim 2, wherein said one or more alternative formats includes PDF.

4. A method as in claim 2, wherein said one or more alternative formats includes HTML.

5. A database server for analyzing and reporting statistical and historical data within a plurality of database systems, comprising:

means for storing collected data and statistics received from said database systems into at least one database repository;

means for providing historical trends-based analysis on said collected data and statistics;

means for converting results of said analysis into one or more alternative formats;

means for transmitting said results to the sender of said collection request; and

means for providing, to said sender of said collection request, additional information about said collected data.

6. A computer readable medium containing computer program instructions for analyzing and reporting statistical and historical data within a plurality of database systems, said computer program instructions containing instructions for:

storing collected data and statistics received from said database systems into at least one database repository;

providing historical trends-based analysis on said collected data and statistics;

converting results of said analysis into one or more alternative formats;

transmitting said results to the sender of said collection request; and

providing, to said sender of said collection request, additional information about said collected data.

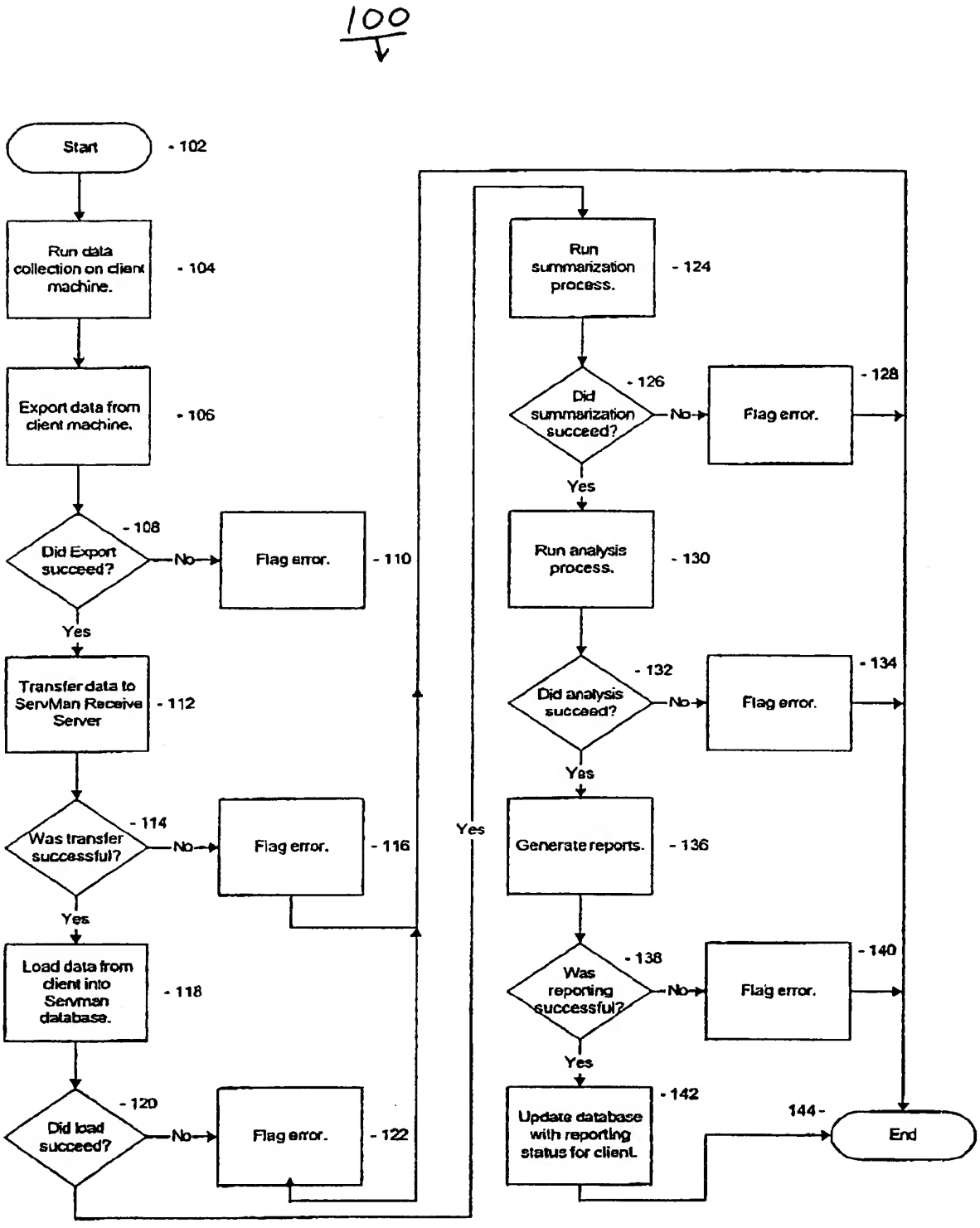

100
Start
102
Run data collection on client machine.
104
Export data from client machine.
106
Did Export succeed?
108
No
Flag error.
110
Yes
Transfer data to ServMan Receive Server
112
Was transfer successful?
114
No
Flag error.
116
Yes
Load data from client into Servman database.
118
Did load succeed?
120
No
Flag error.
122
Yes
Run summarization process.
124
Did summarization succeed?
126
No
Flag error.
128
Yes
Run analysis process.
130
Did analysis succeed?
132
No
Flag error.
134
Yes
Generate reports.
136
Was reporting successful?
138
No
Flag error.
140
Yes
Update database with reporting status for client
142
144
End

FIG. 1

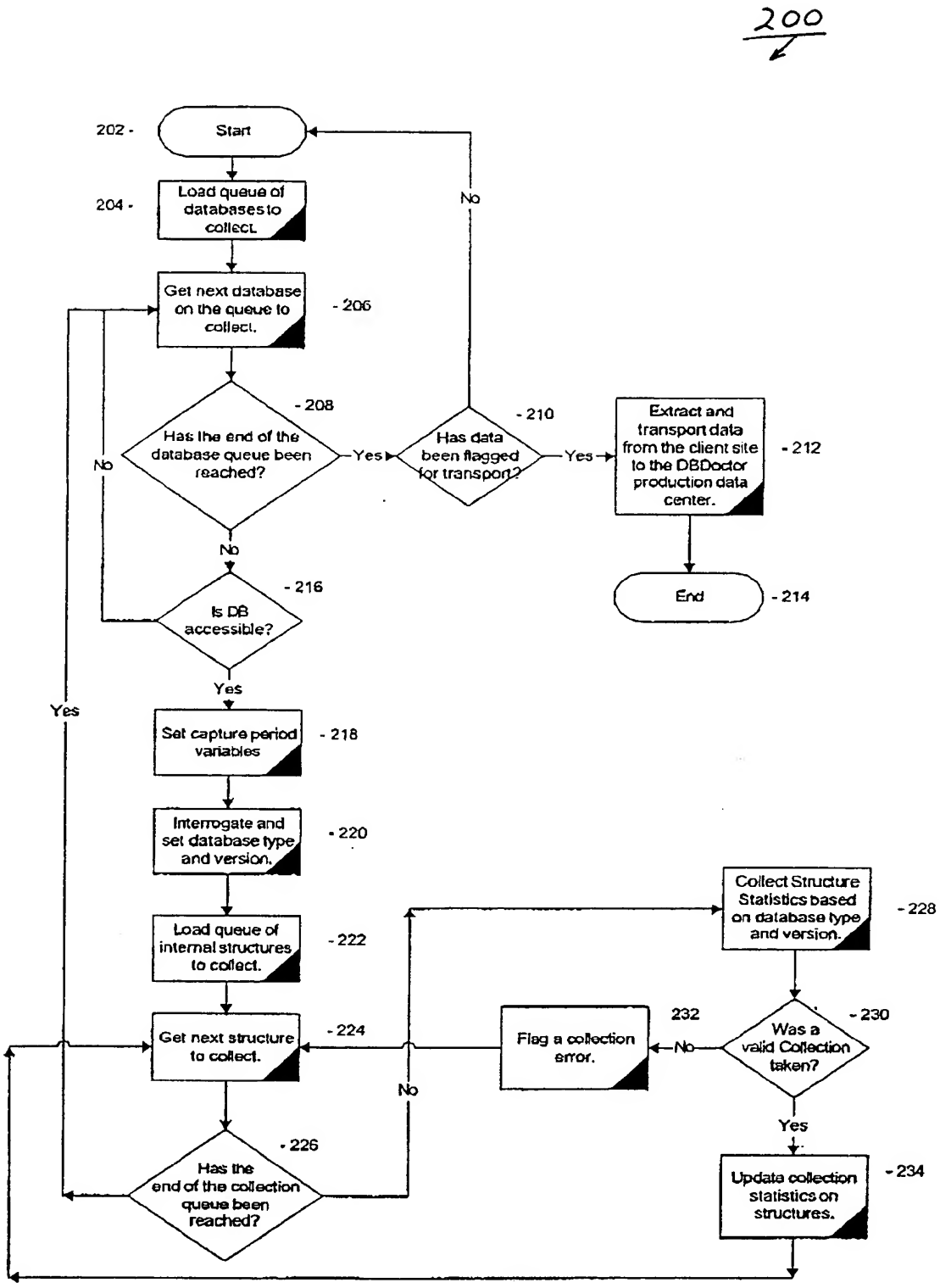

200
202 -
Start
204 -
Load queue of databases to collect.
Get next database on the queue to collect.
- 206
- 208
Has the end of the database queue been reached?
Yes
No
- 210
Has data been flagged for transport?
Yes
No
Extract and transport data from the client site to the DBDoctor production data center.
- 212
End
- 214
- 216
Is DB accessible?
Yes
No
Yes
Set capture period variables
- 218
Interrogate and set database type and version.
- 220
Load queue of internal structures to collect.
- 222
Get next structure to collect.
- 224
- 226
Has the end of the collection queue been reached?
No
Yes
Collect Structure Statistics based on database type and version.
- 228
- 230
Was a valid Collection taken?
232
No
Flag a collection error.
Yes
Update collection statistics on structures.
- 234

Fig. 2

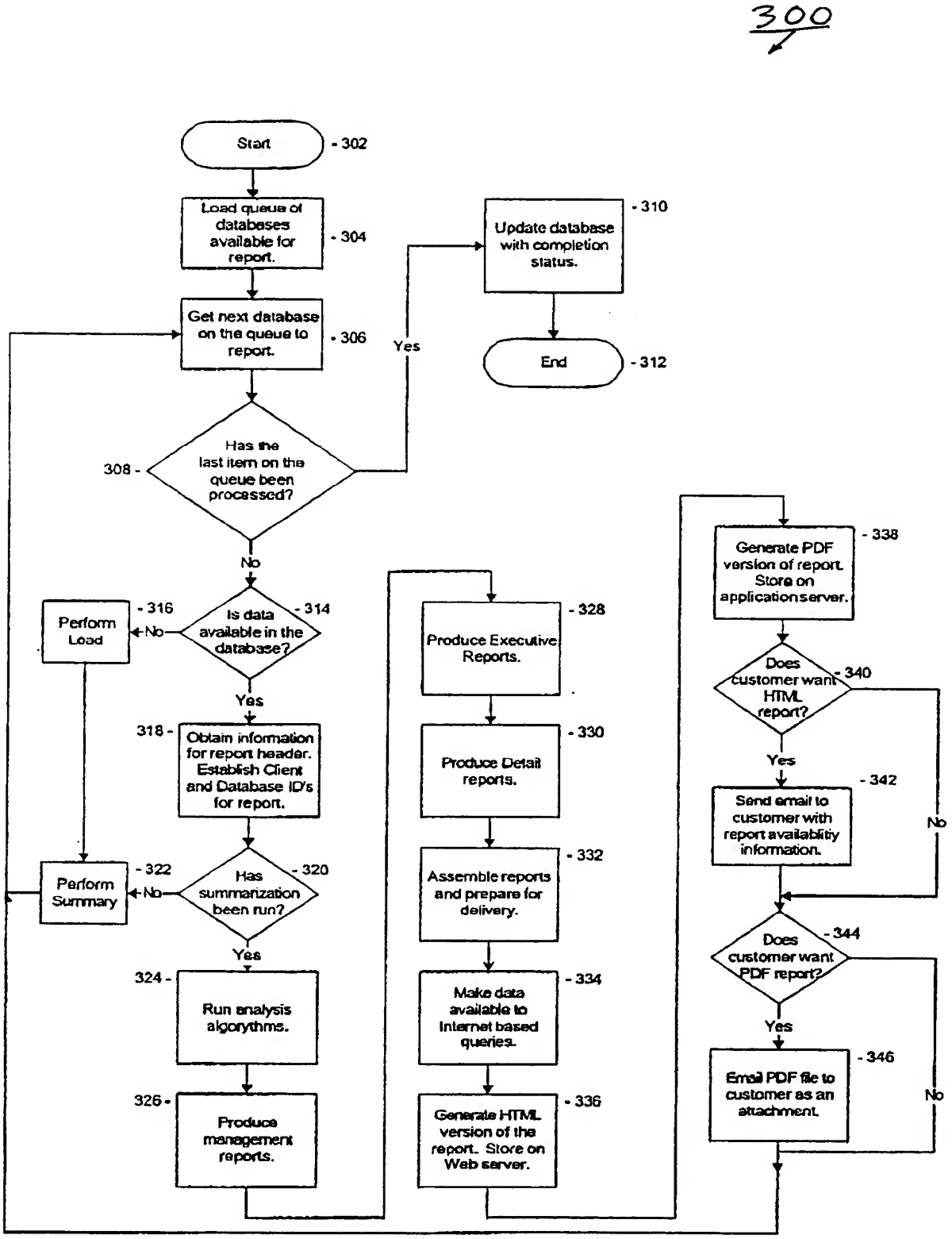

300
Start
302
Load queue of databases available for report.
304
Get next database on the queue to report.
306
Has the last item on the queue been processed?
308
Yes
No
Update database with completion status.
310
End
312
Is data available in the database?
314
No
Perform Load
316
Yes
Obtain information for report header. Establish Client and Database ID's for report.
318
Has summarization been run?
320
No
Perform Summary
322
Yes
Run analysis algorythms.
324
Produce management reports.
326
Produce Executive Reports.
328
Produce Detail reports.
330
Assemble reports and prepare for delivery.
332
Make data available to Internet based queries.
334
Generate HTML version of the report. Store on Web server.
336
Generate PDF version of report. Store on application server.
338
Does customer want HTML report?
340
Yes
No
Send email to customer with report availablitiy information.
342
Does customer want PDF report?
344
Yes
No
Email PDF file to customer as an attachment.
346

Fig. 3

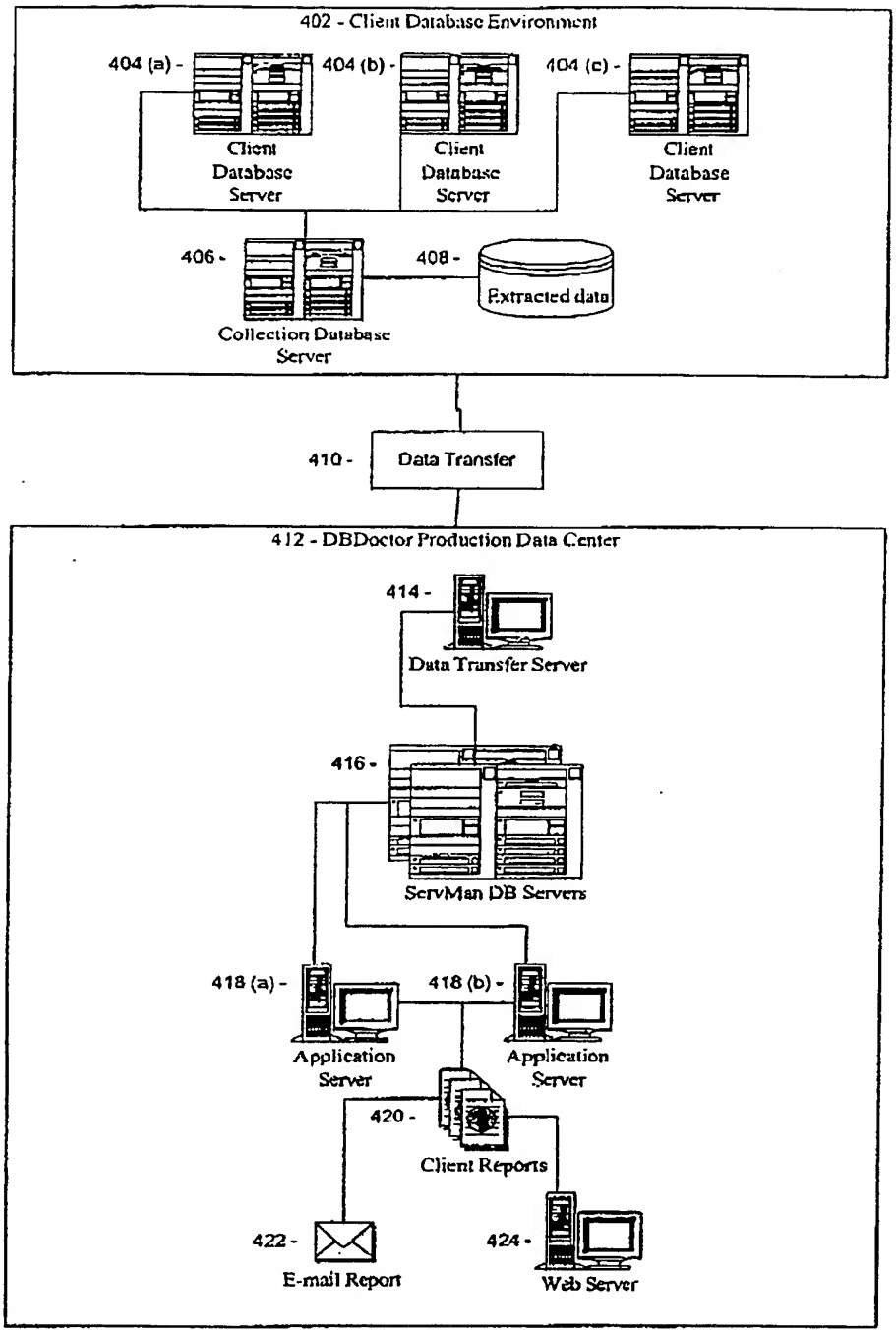

402 - Client Database Environment
404 (a) -
404 (b) -
404 (c) -
Client Database Server
Client Database Server
Client Database Server
406 -
408 -
Extracted data
Collection Database Server
410 -
Data Transfer
412 - DBDoctor Production Data Center
414 -
Data Transfer Server
416 -
ServMan DB Servers
418 (a) -
418 (b) -
Application Server
Application Server
420 -
Client Reports
422 -
E-mail Report
424 -
Web Server

Fig. 4

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep, 1999

Executive Summary

Severity ~512 *Category* ~510

Critical	Physical layout of Oracle should be reviewed.
Serious	Database objects require attention to allow for future growth.
Serious	Physical layout of Oracle should be reviewed.
Warning	Placement of database objects require attention.
Warning	Resources which have been assigned to users should be reviewed.
Warning	Storage parameters should be reviewed.
Notification	Database objects require attention.
Notification	Placement of database objects require attention.

End of Report.

Fig. 5(a)

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep, 1999

Management Report *Threats and Wari*

Severity: Critical ~ 520

Executive Category: Physical layout of Oracle should be reviewed.

Condition Encountered

Physical layout of control files should be reviewed.
Physical layout of redo logs should be reviewed.

Severity: Serious ~ 522

Executive Category: Database objects require attention to allow for future growth.

Condition Encountered

Tables exist which are nearing max extents.

Executive Category: Physical layout of Oracle should be reviewed.

Condition Encountered

Physical layout of control files should be reviewed.
Physical layout of redo logs should be reviewed.

Severity: Warning ~ 524

Fig. 5(b)

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep, 1999

Detail Report

Threats and Warning

Severity: Critical

Management Category: Physical layout of control files should be reviewed.

Status	*Condition Encountered*
☐	ALL control files are on the same mount point /data3.
☐	Control file /data3/oracle/dbs/ora_control1 is on the same mount point as other control fi
☐	Control file /data3/oracle/dbs/ora_control2 is on the same mount point as other control fi

Management Category: Physical layout of redo logs should be reviewed.

Status	*Condition Encountered*
☐	All redo logs have been placed on the same disk.
☐	There are redo log groups that do not have enough log members in them.

Severity: Serious

Management Category: Tables exist which are nearing max extents.

Status	*Condition Encountered*
☐	Table CLIENTS owned by SALES_OWNER is nearing max extents.
☐	Table INVOICE owned by ACCOUNTING_OWNER is nearing max extents.

Fig. 5(c)

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep. 1999

Executive Summary

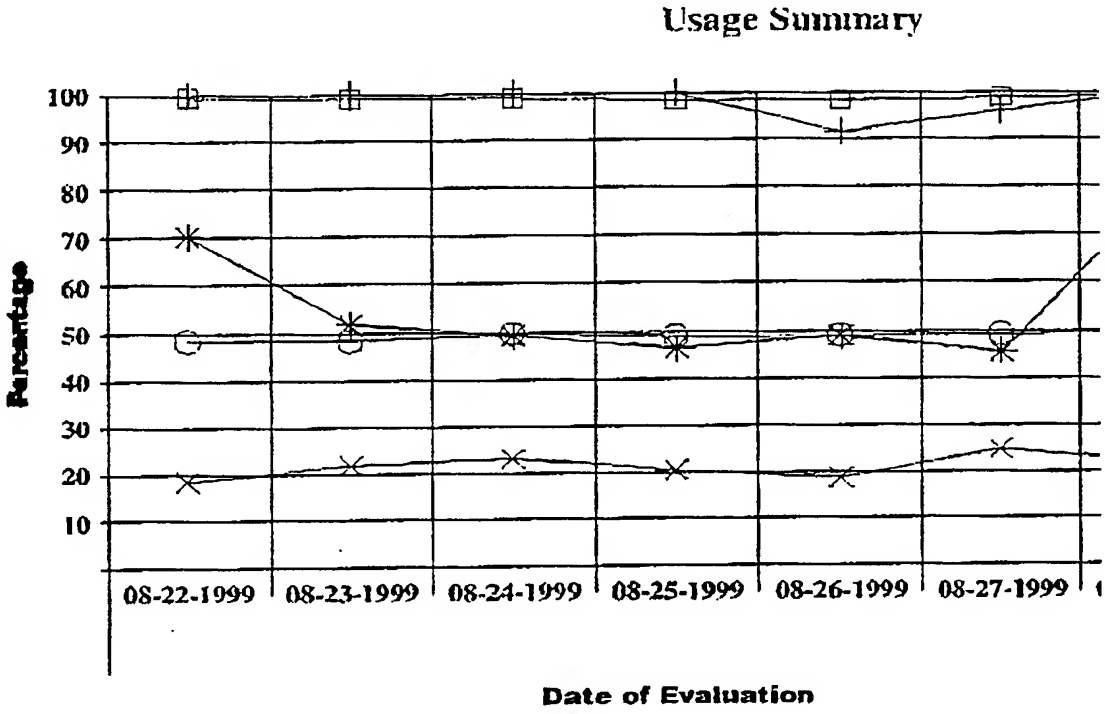


Fig 5(d)

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep. 1999

Management Report

I/O Mount Point Percentages - Sunday, August

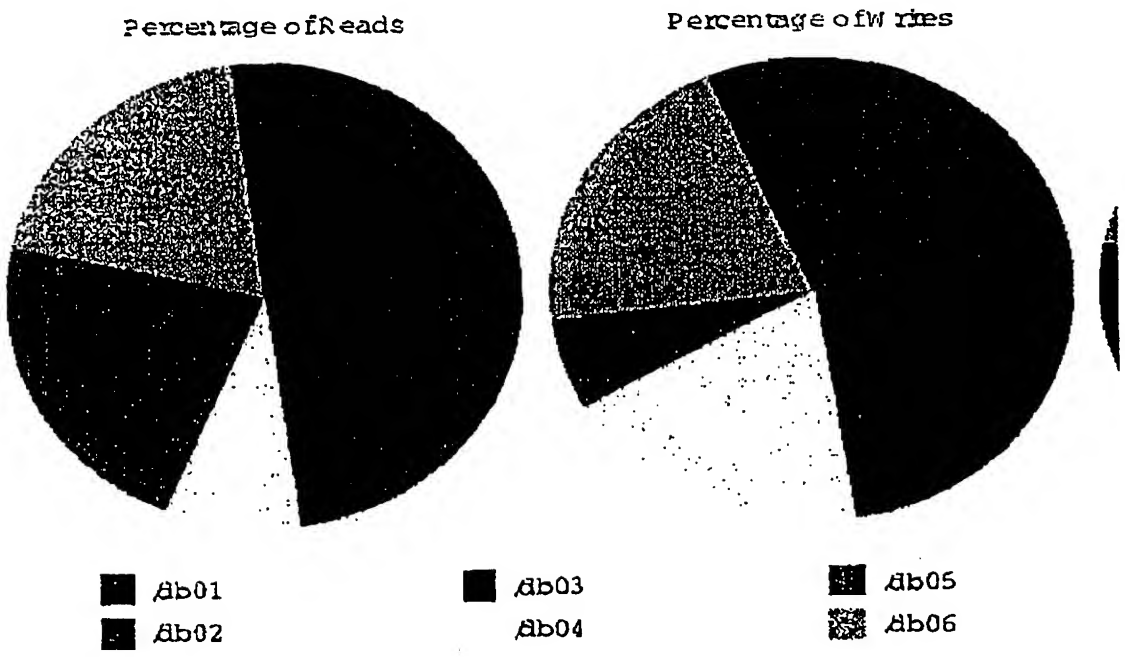


Fig 5(c)

DBDoctor for Oracle

Client Name: TheCorp.com
Database clientdb1
Evaluation Date: 01 Sep. 1999

Detail Report

Evaluation Date	Mount Point	Percentage of Total	Percentage of Reads	Percentage of Writes	Physical
08-22-1999	/db01	25.06	25.29	4.54	6.5
08-22-1999	/db02	16.87	16.84	17.16	4,3
08-22-1999	/db03	8.02	5.87	25.93	1,5
08-22-1999	/db04	9.6	8.45	19.18	2,2
08-22-1999	/db05	20.59	22.3	6.4	5,8
08-22-1999	/db06	19.41	19.29	20.42	5,0
08-22-1999	/db07	2.44	1.97	6.37	51
08-23-1999	/db01	25.58	28.26	3.97	10,8
08-23-1999	/db02	15.65	15.69	15.3	5,9
08-23-1999	/db03	7.28	4.71	28.01	1,8
08-23-1999	/db04	8.41	6.94	20.29	2,6
08-23-1999	/db05	20.43	22.54	3.4	8,6
08-23-1999	/db06	18.88	18.41	22.72	7,0
08-23-1999	/db07	3.78	3.46	6.31	1,3
08-24-1999	/db01	26.8	29.48	4.14	11,2
08-24-1999	/db02	13.47	13.35	14.5	5,1
08-24-1999	/db03	7.82	5.49	27.54	2,1
08-24-1999	/db04	9.55	7.83	24.09	2,9
08-24-1999	/db05	19.03	20.81	3.99	7,9
08-24-1999	/db06	19.24	19.02	21.16	7,2
08-24-1999	/db07	4.08	4.02	4.59	1,5
08-25-1999	/db01	27.9	30.03	5.76	13,2
08-25-1999	/db02	14.8	14.97	13.06	6,5

Page 83 of 122

Fig. 5(E)